The Weighted Möbius Score: A Unified Framework for Feature Attribution

Yifan Jiang

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2023

Committee:
Shane Steinert-Threlkeld
Yonatan Belinkov

Program Authorized to Offer Degree:
Department of Linguistics

University of Washington

Abstract

The Weighted Möbius Score:
A Unified Framework for Feature Attribution

Yifan Jiang

Chair of the Supervisory Committee:
Shane Steinert-Threlkeld
Department of Linguistics

Feature attribution aims to explain the reasoning behind a black-box model's prediction by identifying the impact of each feature on the prediction. Recent work has extended feature attribution to interactions between multiple features. However, the lack of a unified framework has led to a proliferation of methods that are often not directly comparable. This thesis introduces a parameterized attribution framework—the Weighted Möbius Score—and (i) shows that many different attribution methods for both individual features and feature interactions are special cases and (ii) identifies some new methods. By studying the vector space of attribution methods, our framework utilizes standard linear algebra tools and provides interpretations in various fields, including cooperative game theory and causal mediation analysis. We empirically demonstrate the framework's versatility and effectiveness by applying these attribution methods to feature interactions in sentiment analysis and Chain-of-Thought prompting.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I am deeply grateful to my advisor, Shane Steinert-Threlkeld, whose guidance and encouragement have been instrumental in my growth as a researcher. I equally appreciate Yonatan Belinkov for his insightful feedback, which greatly enriched my work. Additionally, I would like to thank my friends, Hao Li and Haoran Liang, for their stimulating discussions. Above all, I am thankful to my family for their unconditional love and support, which served as a strong backbone throughout this journey.

DEDICATION

To Yihan, my parents, and all the people who have helped me along the way.

Chapter 1

INTRODUCTION

Explaining the predictions made by black-box machine learning models, such as neural networks, poses a significant challenge. To address this challenge, feature attribution has become a popular approach, aimed at determining the impact of individual features on a model's prediction. However, the application of these attribution methods for explaining feature interactions, which play a critical role in many real-world scenarios, remains an open problem. For example, in sentiment analysis, the interaction between the words "not" and "bad" results in a prediction that differs from what either word alone would produce. The capability to encode such interactions is believed to be the reason for the success of neural networks [Goodfellow et al., 2016] which highlights the need for a unified framework for feature attribution that can be applied to both individual features and feature interactions.

Recently, there has been a growing interest in extending attribution methods to feature interactions. Several methods have been proposed [Tsang et al., 2017, Sundararajan et al., 2019, Janizek et al., 2020, Tsang et al., 2020], but they are often not directly comparable due to their different assumptions. Empirical studies have produced conflicting results, and it is challenging to determine which method is superior, as the results may depend on the specific task and model being used. Furthermore, ground truth attributions are often not available for real-world tasks, making it difficult to compare these methods empirically.

This thesis presents a unified framework—the *Weighted Möbius Score*—for model-agnostic feature attribution for both individual features and feature interactions (Section 2). Our framework situates feature attribution methods within a vector space, which is then analyzed using standard linear algebraic tools. Our framework also has a natural interpretation in terms of cooperative game theory and causal mediation analysis, providing a unified perspective for

understanding existing attribution methods and developing new ones (Section 3).

Our contributions include:

(1) A unified framework for model-agnostic local feature attribution, based on linear algebra, that can be applied to both individual features and feature interactions.

(2) A theoretical analysis that bridges concepts from feature attribution, cooperative game theory, and causal mediation analysis.

(3) An empirical demonstration of the framework's versatility and effectiveness on real-world tasks such as sentiment analysis and prompt engineering (Section 4).

Chapter 2
THE WEIGHTED MÖBIUS SCORE

In this section, we outline the framework for local feature attribution methods, which are designed to explain the reasoning behind a model's prediction for a single input. Our framework is model-agnostic, meaning that it is applicable to any black-box model and does not rely on any assumptions about the model's architecture, training procedure, or mathematical properties.

Notations We denote the model as a function $f : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X}$ denotes a d-dimensional vector space and $\mathcal{Y}$ represents the output space. We denote the set of all features in $\mathcal{X}$ by $\mathcal{D} = \{1, 2, ..., d\}$. For any input $x \in \mathcal{X}$ and any subset of features $S \subseteq \mathcal{D}$, counterfactual inputs $x_{\setminus S}$ can be constructed by removing the features in S from x. Many techniques have been proposed for removing features. For example, Occlusion [Zeiler and Fergus, 2013] removes features by zeroing them out, while LIME [Ribeiro et al., 2016] replaces the features with a default value. A thorough review of feature removal techniques can be found in Covert et al. [2020].

Definition 1 (Local Attribution Method)**.** Given a model f and an input x, a local attribution method A is a function $A : \mathcal{P}(\mathcal{D}) \to \mathbb{R}$, where $\mathcal{P}(\mathcal{D})$ is the power set of $\mathcal{D}$. By "local", we mean that the attribution method only applies to the neighborhood of x rather than the entire input space $\mathcal{X}$. Some attribution methods only consider individual features and do not consider subsets with cardinality greater than one. These methods can be seen as a special case of A, where $A(S) = 0$ for all S with $|S| > 1$.

Definition 2 (Vector Space of Local Attribution Methods)**.** The space of local attribution methods forms a vector space $\mathcal{A} = \mathbb{R}^{\mathcal{P}(\mathcal{D})}$, where $\mathbb{R}^{\mathcal{P}(\mathcal{D})}$ is the space of functions mapping

from $\mathcal{P}(\mathcal{D})$ to $\mathbb{R}$. The space has point-wise addition and scalar multiplication: $(A+B)(S) := A(S) + B(S)$ and $(cA)(S) := c \cdot A(S)$ for each subset $S \subseteq \mathcal{D}$.

The space has dimension 2^d with a natural basis $\{\mathbf{1}_S : S \subseteq \mathcal{D}\}$, where $\mathbf{1}_S$ is the function that assigns 1 to S and 0 to all other subsets. This representation enables the application of standard linear algebra tools to analyze local attribution methods. The Zeta transform and its inverse, the Möbius Transform, are examples of such tools [Stanley, 2011].

Definition 3 (Zeta Transform and Möbius Transform on $\mathcal{A}$)**.** The Zeta transform and the Möbius transform are linear operators defined on function spaces with a partially ordered set domain. As such, they can be defined on the space of local attribution methods $\mathcal{A}$. The Zeta transform, denoted as ζ, takes a local attribution method A as an input. It outputs a new method such that for any set S, the value of this new method at S is computed as the sum of the values of A at all subsets of S. Mathematically, this is expressed as $\zeta(A)(S) = \sum_{T \subseteq S} A(T)$. The Möbius transform, denoted as μ, likewise outputs a new method from a local attribution method A. However, the value of this new method at a set S is computed by summing up the values of A at all subsets of S, each multiplied by $(-1)^{|S|-|T|}$. Here, $|T|$ represents the cardinality (i.e., the number of elements) of the subset T. The expression $(-1)^{|S|-|T|}$ is known as the Möbius function, which alternates between -1 and 1 based on the difference in the cardinality of the sets S and T. The formula for the Möbius transform is $\mu(A)(S) = \sum_{T \subseteq S} (-1)^{|S|-|T|} A(T)$. An important property of these transforms is that they are inverses of each other. This means applying the Möbius transform to the output of a Zeta transform (or vice versa) on a method A will yield the original method A. In mathematical terms, $\mu(\zeta(A)) = A$ and $\zeta(\mu(A)) = A$. As a generalization of the inclusion-exclusion principle, the Zeta transform and the Möbius transform offer a powerful framework for exploring the properties of local attribution methods.

Definition 4 (Feature Isolation Score)**.** Let f be a model with output space $\mathbb{R}$. [1] Then the

[1] We can map any non-real output space $\mathcal{Y}$ to $\mathbb{R}$ via an appropriate transformation. For example, a binary output space $\mathcal{Y} = \{0, 1\}$ can be mapped to a single real number by calculating the log odds ratio between the positive and negative classes.

feature isolation score can be defined as

$$A_f(S) = f(x_{\setminus \overline{S}}) - f(x_{\setminus \mathcal{D}})$$

This score evaluates the significance of a subset of features S by comparing the predictions of f on two counterfactual inputs: the first with the features outside S removed, and the second with all features removed. Due to its simplicity and ease of implementation, the feature isolation score serves as a useful starting point for the development of more advanced attribution techniques. Similar concepts have been explored in the literature, such as subset extension in Covert et al. [2020].

Definition 5 (Möbius Score)**.** Given the feature isolation score A_f, the Möbius Score $A_{\mu(f)}$ is defined as the Möbius transform of A_f, i.e.:

$$A_{\mu(f)}(S) = \mu(A_f)(S)$$

Alternatively, the Möbius score can be recursively defined as:

$$A_{\mu(f)}(S) = \begin{cases} A_f(S) & \text{if } |S| = 0 \\ A_f(S) - \sum_{T \subset S} A_{\mu(f)}(T) & \text{if } |S| > 0 \end{cases}$$

The Möbius score has the desirable *efficiency* property, which means that the model's prediction for a given input can be completely decomposed into the sum of the Möbius Scores of all feature subsets present in the input. As such, this leaves no contribution unattributed.

Furthermore, the Möbius score satisfies a notion of *identifiability*: the Möbius score for a feature subset identifies the highest-order interaction within that set. For example, in a regression model with interaction terms, this means that the Möbius score of each feature subset is equal to the corresponding terms in the model.

Example 1 (Polynomial Model)**.** Consider a polynomial model $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1^2 + \beta_4 x_2^2 + \beta_5 x_1 x_2$. Assume we remove features by setting them to 0, then the feature isolation

score of each feature subset is as follows:

$$A_y(S) = \begin{cases} 0 & \text{if } S = \emptyset \\ \beta_1 x_1 + \beta_3 x_1^2 & \text{if } S = \{1\} \\ \beta_2 x_2 + \beta_4 x_2^2 & \text{if } S = \{2\} \\ \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1^2 + \beta_4 x_2^2 + \beta_5 x_1 x_2 & \text{if } S = \{1,2\} \end{cases}$$

By applying the Möbius transform, we obtain the Möbius score of each feature subset:

$$A_{\mu(y)}(S) = \begin{cases} A_f(\emptyset) = 0 & \text{if } S = \emptyset \\ A_f(\{1\}) - A_f(\emptyset) = \beta_1 x_1 + \beta_3 x_1^2 & \text{if } S = \{1\} \\ A_f(\{2\}) - A_f(\emptyset) = \beta_2 x_2 + \beta_4 x_2^2 & \text{if } S = \{2\} \\ A_f(\{1,2\}) - A_f(\{1\}) - A_f(\{2\}) + A_f(\emptyset) = \beta_5 x_1 x_2 & \text{if } S = \{1,2\} \end{cases}$$

A more sophisticated example is the Taylor polynomial of a d-th continuous differentiable model f around a baseline input 0. The Möbius score of each feature subset is given by:

$$A_{\mu(\text{Taylor}(f))}(S) = \sum_{I \in \mathcal{I}_S} \frac{D_I f(0)}{I!} x^I$$

Here, $\mathcal{I}_S$ denotes the set of all multi-indices I with $I_i \geq 1$ for all $i \in S$ and $I_i = 0$ for all $i \notin S$. With this notation, $D_I f(0) = \frac{\partial^{|I|} f(0)}{\partial x_1^{I_1} \ldots \partial x_d^{I_d}}$, $I! = \prod_{i \in S} I_i!$ and $x^I = \prod_{i \in S} x_i^{I_i}$.

Definition 6 (Weighted Möbius Score)**.** Given a weight function $\mathbf{w} : \mathcal{P}(\mathcal{D}) \times \mathcal{P}(\mathcal{D}) \to \mathbb{R}$, the *weighted Möbius score* is defined by

$$A_{\mathbf{w}}(S) = \sum_{T \subseteq \mathcal{D}} \mathbf{w}(S,T) A_{\mu(f)}(T)$$

We say that $\mathbf{w}$ (and the corresponding $A_{\mathbf{w}}$) is faithful just in case $\mathbf{w}(S,T) = 0$ if $S \cap T = \emptyset$.

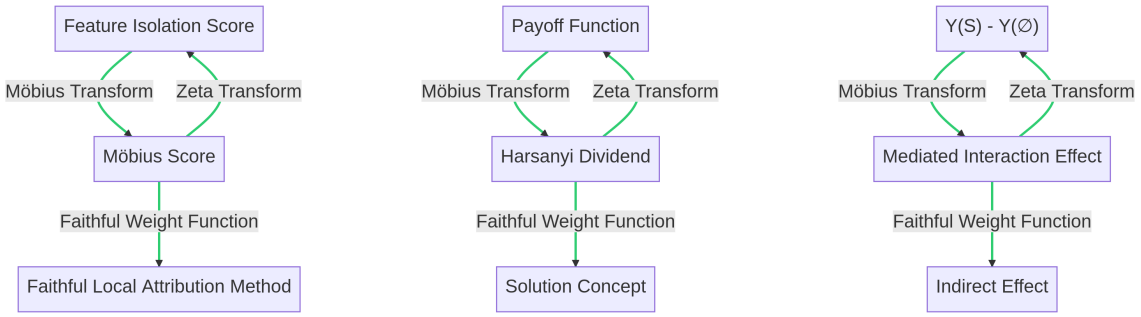
Definition 7 (Faithful Local Attribution Method)**.** A local attribution method A is *faithful* just in case $A = A_{\mathbf{w}}$ for some faithful $\mathbf{w}$.

Intuitively, this means that a faithful local attribution method only attributes interactions between features to those features that are actually involved. Disjoint subsets of features are given a weight of 0, ensuring that the method only attributes importance to relevant features within a given feature subset and not to irrelevant ones, thus remaining faithful to the model. The set of all faithful local attribution methods is a subspace of $\mathcal{A}$, and each element in it is characterized by a weight function $\mathbf{w}$, which can also be interpreted as a linear operator on $\mathcal{A}$. This definition allows for an analysis of existing attribution methods within a unified framework: we will now show that many existing attribution methods are faithful, differing only in the choice of $\mathbf{w}$.

Chapter 3

INTERPRETATION AND CONNECTION TO EXISTING WORK

In this section, we show that many existing attribution methods can be seen as instances of the faithful weighted Möbius score, with different weight functions $\mathbf{w}$. We first focus on a family of methods inspired by cooperative game theory and then on causal mediation analysis. A comparison of the three approaches is shown in Figure 3.1. A summary of the attribution methods discussed in this section (as well as the basic Möbius score from Definition 5) can be found in Table 3.1. Proofs of the results in this section can be found in Appendix A.



(a) Weighted Möbius Score (b) Cooperative Game Theory (c) Causal Mediation Analysis

Figure 3.1: Comparison of the three approaches to feature attribution. These approaches share the same underlying structure, resulting in equivalent frameworks when suitable choices of relevant parameters are made.

Table 3.1: Summary of Attribution Methods. For all values not mentioned, $\mathbf{w}(S,T) = 0$.

Method	$\mathbf{w}(\boldsymbol{S},\boldsymbol{T})$	**Order** [1]
Möbius Score	1 if $S = T$	Up to $\lvert\mathcal{D}\rvert$-order
Shapley Value	$\frac{1}{\lvert T\rvert}$ if $\lvert S\rvert = 1$ and $S \subseteq T$	First-Order
Shapley Interaction Index	$\frac{1}{\lvert T\rvert - \lvert S\rvert + 1}$ if $\lvert S\rvert \leq k$ and $S \subseteq T$	Up to k-th Order
Shapley-Taylor Interaction Index	$\begin{cases} 1 & \text{if } \lvert S\rvert < k \text{ and } S = T \\ \frac{1}{\binom{\lvert T\rvert}{k}} & \text{if } \lvert S\rvert = k \text{ and } S \subseteq T \end{cases}$	Up to k-th Order
Pure Indirect Effect	1 if $\lvert S\rvert = 1$ and $S = T$	First-Order
Total Indirect Effect	1 if $\lvert S\rvert = 1$ and $S \subseteq T$	First-Order
Mediated Interaction Effect [2]	1 if $\lvert S\rvert = 2$ or 3 and $S = T$	Second/Third-Order
ArchAttribute	1 if $\lvert S\rvert = k$ and $T \subseteq S$	k-th Order

[1] "Order" refers to the cardinality of the feature subset that the method can explain.

[2] A generalization to higher-order interactions has been proposed in this section.

3.1 Cooperative Game Theory

Cooperative Game Cooperative game theory is a branch of game theory that studies how players can work together to achieve a common goal. Given a model f and an input x, we can model the attribution problem as a cooperative game $G = (N, v)$ where $N = \mathcal{D}$ is the set of players and $v = A_f$ is the payoff function. In this game, each player $i \in N$ represents a feature in the input x and each coalition $S \subseteq N$ represents a subset of features. The goal of the game is for the players to form a coalition that maximize their joint payoff, or equivalently, to find a subset of features that maximizes the model output relative to the baseline input $x_{\setminus D}$.

Solution Concept The grand coalition N is often assumed to yield the maximum payoff in the game G. An allocation of the payoff of the grand coalition among the players is referred to as a solution concept of the game G. The problem of feature attribution can be viewed as finding a solution concept of the game G that satisfies certain desirable properties. Consequently, each solution concept of the game G can be seen as a local attribution method.

Harsanyi Dividend The Harsanyi dividend is a concept introduced by Harsanyi [1958] to analyze solution concepts. It can be defined recursively as follows:

$$d_v(S) = \begin{cases} v(S) & \text{if } |S| = 0 \\ v(S) - \sum_{T \subset S} d_v(T) & \text{if } |S| > 0 \end{cases}$$

This concept quantifies the surplus of a coalition S that cannot be attributed to the surplus of its sub-coalitions. It is important to note that **the Harsanyi dividend is equivalent to the Möbius score when** $v = A_f$. This equivalence provides a game-theoretic interpretation of the Möbius score and supports the *identifiability* property.

Shapley Value The Shapley value [Shapley, 1953] is a solution concept known for its unique satisfaction of certain fairness axioms, making it widely used in feature attribution.

Many existing attribution methods can be viewed as approximations of the Shapley value [Covert et al., 2020], including LIME [Ribeiro et al., 2016] and SHAP [Lundberg and Lee, 2017]. The Shapley value of a player i can be defined using the Harsanyi dividend as follows [Harsanyi, 1958]:

$$\phi(i) = \sum_{T \subseteq N : i \in T} \frac{1}{|T|} d_v(T)$$

The Shapley value allocates to each player a weighted sum of the Harsanyi dividend for all coalitions that include the player. The weight function, $\mathbf{w}(S, T)$, is given by $\frac{1}{|T|}$ if $\{i\} = S \subseteq T$ and 0 otherwise. This function suggests that the Harsanyi dividend for a coalition T is divided equally among its players, providing a non-axiomatic rationale for the fairness of the Shapley value. Furthermore, because the Shapley value only considers coalitions that contain the player, it is *faithful*.

Shapley (-Taylor) Interaction Indices Interaction indices, generalizing the Shapley value to higher-order interactions, assign a value to each coalition S with a size up to k (i.e., $I(S) = 0$ for $|S| > k$) while satisfying axioms analogous to the Shapley value. The Shapley interaction index [Grabisch and Roubens, 1999] and the Shapley-Taylor interaction index [Sundararajan et al., 2019] are two popular variants. The Shapley interaction index can be defined using the Harsanyi dividend as follows [Grabisch and Roubens, 1999]:

$$I_{\text{SII}}(S) = \sum_{T \subseteq \mathcal{D} : S \subseteq T} \frac{1}{|T| - |S| + 1} d_v(T)$$

The weight function, $\mathbf{w}(S, T)$, is given by $\frac{1}{|T|-|S|+1}$ if $|S| \leq k$ and $S \subseteq T$, and 0 otherwise. This implies that the Harsanyi dividend of T is shared equally between S (considered as a single player) and the remaining players in $T \setminus S$. The Shapley-Taylor interaction index can be also derived from the Harsanyi dividend as follows [Sundararajan et al., 2019, Hamilton et al., 2021]:

$$I_{\text{STI}}(S) = \begin{cases} d_v(S) & \text{if } |S| < k \\ \sum_{T \subseteq \mathcal{D} : S \subseteq T} \binom{|T|}{k}^{-1} d_v(T) & \text{if } |S| = k \end{cases}$$

For $|S| < k$, $I_{\text{STI}}(S)$ is equivalent to the Harsanyi dividend of S, making the weight function $\mathbf{w}(S,T)$ equal to 1 if $S = T$ and 0 otherwise. If $|S| = k$, the index represents the weighted average of the Harsanyi dividend for all coalitions containing S. In this case, the weight function $\mathbf{w}(S,T)$ is given by $\binom{|T|}{k}^{-1}$, which is the inverse of the number of sub-coalitions of size k within T. Thus the Harsanyi dividend of a coalition T is distributed evenly among all sub-coalitions of size k within T.

3.2 Causal Mediation Analysis

Causal Mediation Model Causal mediation analysis is a causal inference method that studies the causal relationship between variables, specifically focusing on the mediating effect through certain intermediate variables, or mediators. Given a model f and an input x, we can model the attribution problem as a causal mediation model (X, M, Y). Within this framework, the treatment X can be either the original input x or the counterfactual input $x_{\setminus\mathcal{D}}$. The mediators M correspond to the features in $\mathcal{D}$ that may be affected by the treatment, while the outcome Y represents the model's prediction, given the treatment and the mediators. We assume that M completely mediates the effect of X on Y, meaning that Y is independent of X when conditioned on M. This assumption deviates from the standard approach in causal mediation analysis, where the treatment is assumed to have a direct effect on the outcome, not through mediators. However, it is reasonable in the context of feature attribution, where the input is determined by its features. Hence, Y can be expressed as a function depending only on M: $Y(S) = f(x_{\setminus\overline{S}})$ for all $S \subseteq M$. This function represents the model's prediction for the counterfactual input $x_{\setminus\overline{S}}$, where the features in S have been removed.

Decomposition of the Total Effect The total effect (TE) measures the effect of a treatment on an outcome, without considering any mediators, which can be expressed as $Y(M) - Y(\emptyset)$ using the causal mediation model defined above. It can be decomposed into direct and indirect effects [Pearl, 2001], where direct effects are not mediated by any variables, and indirect effects are transmitted through one or more mediators. There is no direct effect

when the treatment effect is completely mediated by the mediators, such as in the model defined above. Decomposing the total effect into indirect effects is useful in understanding the causal mechanisms of a treatment, and feature attribution can be seen as finding a decomposition of the total effect that highlights the importance of each mediator.

Mediated Interaction Effect The mediated interaction effect (MI) is an indirect effect that quantifies the interaction effect between the mediator and the treatment or among multiple mediators. Initially proposed by VanderWeele [2013], this concept has been extended to accommodate multiple mediators [Bellavia and Valeri, 2018, Taguri et al., 2018, Gao et al., 2022]; but current definitions only apply to models with two or three mediators. To address this limitation, we propose a general definition that can be applied to any number of mediators:

$$\mathrm{MI}(S) = \sum_{T \subseteq S} (-1)^{|S|-|T|} (Y(T) - Y(\emptyset))$$

where $S \subseteq M$ represents a subset of mediators. Our definition is consistent with the original definition when $|S| \leq 3$. Moreover, **the Mediated Interaction Effect is equivalent to the Möbius Score when** $Y(S) = f(x_{\setminus S})$. In the next section, we will empirically demonstrate that MI, not having been used for feature attribution, can effectively identify interactions encoded in neural networks.

Pure and Total Indirect Effect The pure indirect effect (PIE) and the total indirect effect (TIE) are two of the most commonly used indirect effects in the literature of causal inference. PIE has been extensively applied to the interpretation of neural networks [Vig et al., 2020, Finlayson et al., 2021], while TIE has found its application in [Ban et al., 2022]. Both effects focus on the effect of a single mediator on the outcome Y. The pure indirect effect can be derived from the mediated interaction effect as follows:

$$\mathrm{PIE}(i) = \mathrm{MI}(\{i\})$$

PIE measures the impact of mediator i on the outcome Y when all other mediators remain constant at their counterfactual values. The weight function, $\mathbf{w}(S,T) = 1$ if $|S| = 1$ and $S = T$, and 0 otherwise. In contrast, the total indirect effect can be derived as follows:

$$\text{TIE}(i) = \sum_{T \subseteq \mathcal{D}: i \in T} \text{MI}(T)$$

TIE measures the collective impact of all mediators in $\mathcal{D}$ on the outcome Y, transmitted through mediator i. The weight function for TIE, $\mathbf{w}(S,T) = 1$ if $|S| = 1$ and $S \subseteq T$, and 0 otherwise. The primary distinction between the two indirect effects lies in their treatment of interactions. PIE accounts for only the main effect of the mediator, while TIE considers both the main effect and all potential interactions between the mediator and other mediators.

3.3 Other Related Work

Archipelago Archipelago [Tsang et al., 2020] is a framework designed to extend attribution methods to feature interactions. It comprises two components: an interaction attribution measure, ArchAttribute, and an interaction detector, ArchDetect. Both components can be examined within our framework. ArchAttribute can be represented in terms of the Möbius Score as follows:

$$\phi(S) = \sum_{T \subseteq S} A_{\mu(f)}(T)$$

This definition is essentially the Zeta transform of the Möbius Score, which is equivalent to the Feature Isolation Score. The weight function, $\mathbf{w}(S,T) = 1$ if $T \subseteq S$ and 0 otherwise. ArchDetect, on the other hand, is designed to detect interactions between two features, and can be represented as $\overline{\omega}_{i,j} = \frac{1}{2h_i^2 h_j^2}((\sum_{T \subseteq \mathcal{D}: i,j \in T} A_{\mu(f)}(T))^2 + (A_{\mu(f)}(\{i,j\}))^2)$. Here, $h_i = |x_i - (x_{\setminus \mathcal{D}})_i|$ and $h_j = |x_j - (x_{\setminus \mathcal{D}})_j|$. ArchDetect differs from other methods in that it employs a non-linear function of the Möbius score, rather than a weighted version. Despite this difference, it is clear from the formula that ArchDetect evaluates the combined impact of all potential interactions involving i and j.

Gradient-based Attribution Methods Gradient-based attribution methods have emerged as a popular class of techniques for explaining the predictions of machine learning models. These methods analyze the gradient of a model's output with respect to its input features and have been used to identify important features and interactions. Several individual feature methods, including Integrated Gradients [Sundararajan et al., 2019], SmoothGrad [Smilkov et al., 2017], and DeepLIFT [Shrikumar et al., 2017], and extensions to feature interactions, such as Integrated Hessians [Janizek et al., 2020], have been proposed. However, **these methods depend on certain assumptions about the model's mathematical properties, which may not always hold.** Prior research [Montavon et al., 2015, Ancona et al., 2017] has investigated various approaches to unify gradient-based attribution methods. More recently, Deng et al. [2023] proposed a framework that unifies the Harsanyi dividend and gradient-based attribution methods using Taylor expansion, which provides a potential direction to bridge our framework and gradient-based attribution methods.

Chapter 4

APPLICATIONS

In this section, we first demonstrate the application of our framework to design new attribution methods for causal mediation analysis in sentiment analysis. We then show how our framework can be used to compare existing attribution methods in a black-box prompt engineering setting. [1]

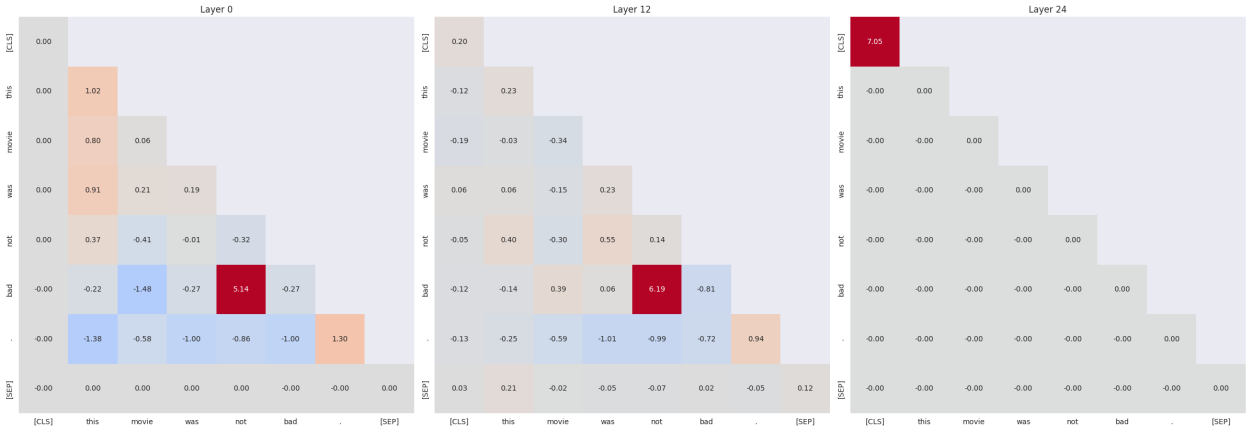
4.1 Designing New Attribution Methods: Sentiment Analysis

Task Definition We focus on sentiment analysis, which classifies text into positive or negative sentiment. Our goal is to understand the contribution of each word's hidden representations and their interactions within the model's decision-making process. We evaluate our methods using the SST-2 dataset [Socher et al., 2013] and employ BERT-large [Devlin et al., 2019] as our base model, obtaining fine-tuned weights [2] from the HuggingFace model hub [Wolf et al., 2019]. We convert the model's binary output into a continuous score using log odds ratio and analyze the model's hidden representations using the causal mediation analysis framework. We employ PIE as the individual attribution measure and the second-order MI as the interaction attribution measure. To the best of our knowledge, this is the first application of MI in the feature attribution context. We randomly sample 100 examples from the validation set and compute layer-wise attribution scores for each example. We conducted the experiments on Google Colab using a single NVIDIA Tesla T4 GPU.

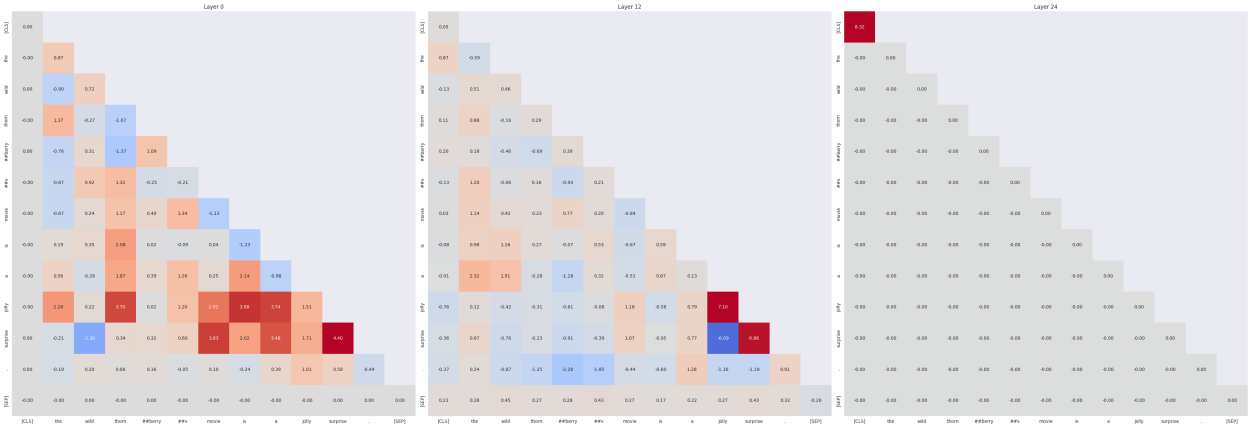
---

[1] Code for all experiments is available at `https://github.com/1fanj/WMS`.

[2] `https://huggingface.co/assemblyai/bert-large-uncased-sst2`



(a) "This movie was not bad" (not from SST-2)



(b) "The wild thornberrys movie is a jolly surprise"

Figure 4.1: Layer-wise MI and PIE scores for the BERT-large model on two example sentences. Three layers are depicted in each graph: the first layer (Layer 0), the middle layer (Layer 12), and the final layer (Layer 24). The diagonal of each matrix represents the PIE scores, while the off-diagonal represents the MI scores. Colors represent the sign and magnitude of the attribution scores, with redder shades indicating more positive scores and bluer shades indicating more negative scores.

Layer-wise MI and PIE Figure 4.1 illustrates layer-wise MI and PIE scores for two sentences. We observe stronger MI scores in lower layers and more pronounced PIE scores in higher layers, suggesting lower layers are more sensitive to word interactions, which higher layers focus more on individual words. In the final layer, nearly all effects concentrate in the CLS token, which is used for the final classification decision. We note that for both examples the CLS token has a high attribution to the positive sentiment, consistent with the model's prediction. In Figure 4.1a, we notice a strong mediated interaction between "not" and "bad" with positive scores, despite both words having negative PIE scores, which is consistent with our expectations. Moreover, in Figure 4.1b, we identify a strong interaction between "jolly" and "surprise" with negative scores, even though both words have positive PIE scores. This counter-intuitive phenomenon, referred to as "saturation" in Janizek et al. [2020], arises when interacting words share the same sentiment polarity as the model's prediction.

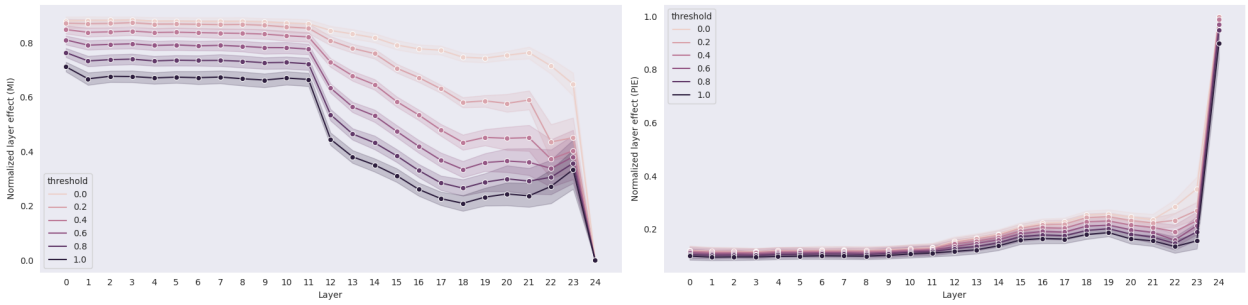


Figure 4.2: Normalized layer effect for the BERT-large model across all 100 examples from the SST-2 dataset. The x-axis represents the layer index, and the y-axis displays the normalized layer effect. Shaded regions show 95% confidence intervals. Thresholds range from 0 to 1 in increments of 0.2. Left plot: normalized MI; Right plot: normalized PIE.

Normalized Layer Effect We calculate the normalized layer effect for MI and PIE, defined as the average proportion of the total magnitude (exceeding a threshold) of a specific type of effect to the total magnitude of all effects. Figure 4.2 visualizes the normalized layer

effect for MI and PIE. We find that normalized MI scores are higher in lower layers and gradually decrease from layer 12 to layer 24 until they reach zero, whereas normalized PIE scores exhibit the opposite trend. The trends are statistically significant, as indicated by the narrow confidence intervals. This finding further supports our previous observation on individual examples that the effects increasingly concentrate on individual tokens as the layers deepen. These results demonstrate the effectiveness of the new attribution methods in revealing the different contributions and interactions of words in various layers of the model, which provides insights into the model's internal decision-making process.

Discussion In this section, we explore sentiment analysis as an application of our framework. Through our newly developed attribution method, we demonstrate how the BERT-large model encodes and utilizes lexical relationships to make predictions. The layer-wise analysis further exposes a non-uniformity in the model's decision-making process across different layers, aligning with previous findings that different layers of the model encode disparate types of information [Tenney et al., 2019, Rogers et al., 2020]. Specifically, previous work has shown that lower and middle layers of the model encode more syntactic/semantic information, whereas final layers encode more task-specific information. This may explain why we observe stronger MI scores in lower layers but stronger PIE scores in higher layers: the model's decision-making process shifts from general word interactions to task-specific tokens (e.g., CLS token) as the layers deepen. We suspect that this is because the model may have incorporated the complex word interactions learned in lower layers into representations of these task-specific tokens. For future work, we plan to conduct more empirical evaluations of our attribution methods on other NLP tasks and datasets. In particular, we are interested in exploring faithfulness evaluations on datasets such as e-XNLI [Zaman and Belinkov, 2022], which is specifically designed to evaluate attribution methods in a multilingual setting.

4.2 Comparing Existing Attribution Methods: Prompt Engineering

Task Definition We focus on prompt engineering, a task aiming to enhance a language model's performance on a downstream task by supplying prompts to the model. One well-known method is Chain of Thought (CoT) [Wei et al., 2022], which uses a sequence of demonstrations to guide the model's reasoning process. Our objective is to understand the contribution of each sentence or word in the demonstrations to the model's performance. We evaluate our methods on the last letter concatenation task proposed in Wei et al. [2022], in which the model concatenates the last letter of each word in a given name. OpenAI's ChatGPT API (gpt-3.5-turbo) [Brown et al., 2020, OpenAI, 2023] [3] is used to obtain the model's predictions with the temperature set to 0 to minimize randomness. We select the first 100 examples from the dataset [4] and compute the attribution scores on a one-shot CoT prompt. We construct the input as follows:

User: "Take the last letters of the words in "Bill Gates" and concatenate them."

Assistant: {Demonstrations} + "The answer is ls"

User: "{Question}"

where {Demonstrations} is a sequence of demonstrations which we vary in our experiments, and {Question} is the question we ask the model, e.g., "Take the last letters of the words in "Waldo Schmidt" and concatenate them.". If the correct concatenation appears in the model's response, we consider the model's prediction to be correct. We compute the Möbius score and four additional attribution scores using the respective weight functions: the Shapley value, the Shapley interaction index, the total indirect effect, and ArchAttribute.

---

[3] Although the results may not be replicable due to the model's closed-source nature, the main focus of this experiment is the comparative analysis of attribution methods, which remains relevant and applicable.

[4] `https://github.com/jasonwei20/chain-of-thought-prompting`

Table 4.1: Sentence-level attribution scores for the last letter concatenation task.

Sentences	**Möbius**	**Shapley**	**SII**	**TIE**	**ArchAttribute**
#1	1.000	0.407	0.407	0.000	1.000
#2	0.987	0.400	0.400	0.000	0.987
#3	0.571	0.193	0.193	0.000	0.571
#1, #2	-0.987	0.000	-0.708	0.000	1.000
#1, #3	-0.571	0.000	-0.292	0.000	1.000
#2, #3	-0.558	0.000	-0.279	0.000	1.000
#1, #2, #3	0.558	0.000	0.558	0.000	1.000

Sentence-level Attribution Table 4.1 presents the sentence-level attribution scores for the last letter concatenation task. The demonstrations consist of three sentences: 'The last letter of "Bill" is "l".' (#1), 'The last letter of "Gates" is "s".' (#2), and 'Concatenating them is "ls".' (#3). Attribution scores are averaged across 77 out of 100 examples where the model's prediction is incorrect without the demonstrations. Our framework yields several insights into each method's behavior: (1) The Möbius score for all sentence pairs is negative, indicating the saturation phenomenon. (2) TIE is entirely uninformative because it magnifies the interaction effects by attributing them to each involved sentence without any normalization. (3) Both the Shapley value and the Shapley interaction index assign nearly equal importance to the first two sentences, as they distribute the interaction effects uniformly across the involved sentences. (4) The Shapley interaction index attributes nearly equal importance to the pair #1, #2 and the pair #2, #3 because it considers the pair as a whole and then distributes the interaction effect uniformly across the involved sentences. (5) The ArchAttribute score matches the model's accuracy, as it is exactly the feature isolation score, which is the difference between the model's outputs when only considering sentences of interest and when excluding the demonstrations entirely. These insights demonstrate the usefulness of our framework in

comparing attribution methods, thereby facilitating a deeper understanding of their strengths and weaknesses.

Table 4.2: Phrase-level attribution scores for the last letter concatenation task.

Phrases	**Möbius**	**Shapley**	**SII**	**TIE**	**ArchAttribute**
NP	0.605	0.206	0.206	0.000	0.605
PP	0.855	0.331	0.331	0.000	0.855
VP	1.000	0.450	0.450	0.000	1.000
NP, PP	-0.566	0.000	-0.276	0.000	0.895
NP, VP	-0.618	0.000	-0.329	0.000	0.987
PP, VP	-0.868	0.000	-0.579	0.000	0.987
NP, PP, VP	0.579	0.000	0.579	0.000	0.987

Phrase-level Attribution Table 4.2 presents the phrase-level attribution scores for the last letter concatenation task. We compute the attribution scores for the noun phrase (NP), prepositional phrase (PP), and verb phrase (VP) in the first sentence of the demonstrations, which are 'The last letter', 'of "Bill"', and 'is "l"', respectively. Similarly, attribution scores are averaged across 76 [5] examples where the model fails without the demonstrations. Observations from Table 4.2 are similar to those from Table 4.1. Additionally, we notice that each individual phrase has a significant influence on the model's prediction, as evidenced by the positive Möbius Scores assigned to each phrase and the negative scores assigned to each pair of phrases, indicating that the presence of key phrases alone can boost performance. This suggests that the CoT prompt's effectiveness may not arise from the step-by-step reasoning process guiding the model, but instead from emphasizing key phrases that enhance its performance.

[5]The number of examples differs from sentence-level attribution due to prediction randomness.

Discussion In this section, we examine different attribution methods' behavior in the context of the CoT prompting. Our framework suggests that game theory-based attribution methods are more suitable than TIE, which we find to be uninformative in this setting. We also discover that the CoT prompt's effectiveness isn't necessarily tied to the logical reasoning it promotes, but to the presence of key phrases. These phrases significantly influence the model's prediction, irrespective of order or logical structure. This aligns with Min et al. [2022], Wang et al. [2022], indicating that in-context learning can be achieved by invalid demonstrations. Understanding the effectiveness of in-context learning is a long-term goal, and we believe our framework can serve as a useful tool for achieving it.

Chapter 5

CONCLUSION

In this paper, we propose a novel model-agnostic framework for understanding the behavior of local feature attribution methods. Our framework introduces the *weighted Möbius score*, which is a principled measure for quantifying the interaction effects between features. We show that this framework can be interpreted in various fields, including cooperative game theory and causal mediation analysis, thereby providing a unified view of feature attribution methods. We demonstrate our framework's usefulness by designing a new attribution method tailored to causal mediation analysis and comparing various feature attribution methods in a fully black-box setting. Our framework can be extended to other attribution methods and applications, which we leave for future exploration. Improving computational efficiency, currently a bottleneck of our framework due to the exponential complexity of the Möbius Score, is also a potential direction for future work.

BIBLIOGRAPHY

Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus H. Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. In *International Conference on Learning Representations*, 2017.

Pangbo Ban, Yifan Jiang, Tianran Liu, and Shane Steinert-Threlkeld. Testing pre-trained language models' understanding of distributivity via causal mediation analysis. In *BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, 2022.

Andrea Bellavia and Linda Valeri. Decomposition of the total effect in the presence of multiple mediators and interactions. *American journal of epidemiology*, 187 6:1311–1318, 2018.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *ArXiv*, abs/2005.14165, 2020.

Ian Covert, Scott M. Lundberg, and Su-In Lee. Explaining by removing: A unified framework for model explanation. *J. Mach. Learn. Res.*, 22:209:1–209:90, 2020.

Huiqi Deng, Na Zou, Mengnan Du, Weifu Chen, Guo-Can Feng, Ziwei Yang, Zheyang Li, and Quanshi Zhang. Understanding and unifying fourteen attribution methods with taylor interactions. *ArXiv*, abs/2303.01506, 2023.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805, 2019.

Matthew Finlayson, Aaron Mueller, Stuart M. Shieber, Sebastian Gehrmann, Tal Linzen, and Yonatan Belinkov. Causal analysis of syntactic agreement mechanisms in neural language models. *ArXiv*, abs/2106.06087, 2021.

Xin Gao, Li Li, and Lijuan Luo. Decomposition of the total effect for two mediators: A natural mediated interaction effect framework. *Journal of causal inference*, 10:18 – 44, 2022.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. URL `http://www.deeplearningbook.org`.

Michel Grabisch and Marc Roubens. An axiomatic approach to the concept of interaction among players in cooperative games. *International Journal of Game Theory*, 28:547–565, 1999.

Mark Hamilton, Scott M. Lundberg, Lei Zhang, Stephanie Fu, and William T. Freeman. Axiomatic explanations for visual search, retrieval, and similarity learning. In *International Conference on Learning Representations*, 2021.

John C. Harsanyi. A bargaining model for the cooperative n-person game. 1958.

Joseph D. Janizek, Pascal Sturmfels, and Su-In Lee. Explaining explanations: Axiomatic feature interactions for deep networks. *J. Mach. Learn. Res.*, 22:104:1–104:54, 2020.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *ArXiv*, abs/1705.07874, 2017.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *Conference on Empirical Methods in Natural Language Processing*, 2022.

Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *ArXiv*, abs/1512.02479, 2015.

OpenAI. Introducing chatgpt, 2023. URL `https://openai.com/blog/chatgpt`.

Judea Pearl. Direct and indirect effects. *Probabilistic and Causal Inference*, 2001.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, 8: 842–866, 2020.

Lloyd S. Shapley. 17. a value for n-person games. 1953.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, 2017.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda B. Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *ArXiv*, abs/1706.03825, 2017.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, A. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Conference on Empirical Methods in Natural Language Processing*, 2013.

Richard P. Stanley. Enumerative combinatorics: Volume 1. 2011.

Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. The shapley taylor interaction index. In *International Conference on Machine Learning*, 2019.

Masataka Taguri, J Featherstone, and Jing Cheng. Causal mediation analysis with multiple causally non-ordered mediators. *Statistical Methods in Medical Research*, 27:19 – 3, 2018.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. Bert rediscovers the classical nlp pipeline. In *Annual Meeting of the Association for Computational Linguistics*, 2019.

Michael Tsang, Dehua Cheng, and Yan Liu. Detecting statistical interactions from neural network weights. *ArXiv*, abs/1705.04977, 2017.

Michael Tsang, Sirisha Rambhatla, and Yan Liu. How does this interaction affect me? interpretable attribution for feature interactions. *ArXiv*, abs/2006.10965, 2020.

Tyler J. VanderWeele. A three-way decomposition of a total effect into direct, indirect, and interactive effects. *Epidemiology*, 24 2:224–32, 2013.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart M. Shieber. Investigating gender bias in language models using causal mediation analysis. In *Neural Information Processing Systems*, 2020.

Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. Towards understanding chain-of-thought prompting: An empirical study of what matters. *ArXiv*, abs/2212.10001, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903, 2022.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771, 2019.

Kerem Zaman and Yonatan Belinkov. A multilingual perspective towards the evaluation of attribution methods in natural language inference. *ArXiv*, abs/2204.05428, 2022.

Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, 2013.

Appendix A

PROOFS

In this appendix, we provide complete proofs for the results presented in Section 3. It is important to note, however, that (i) proofs related to game-theoretic methods have been excluded as they are well-established in the literature and (ii) the definitions used in these proofs may differ from their original presentations in the literature due to the different notations used in this thesis.

Pure Indirect Effect The pure indirect effect for a mediator i can be represented as:

$$\text{PIE}(i) = \text{MI}(\{i\})$$

Proof. We start from the definition of the pure indirect effect:

$$\text{PIE}(i) = Y(\{i\}) - Y(\emptyset)$$

Adding and subtracting $Y(\emptyset)$ from the right-hand side leads to:

$$\text{PIE}(i) = Y(\{i\}) - Y(\emptyset) - (Y(\emptyset) - Y(\emptyset)) = \text{MI}(\{i\})$$

□

Total Indirect Effect The total indirect effect for a mediator i can be represented as:

$$\text{TIE}(i) = \sum_{T \subseteq \mathcal{D}: i \in T} \text{MI}(T)$$

Proof. We start from the definition of the total indirect effect:

$$\text{TIE}(i) = Y(\mathcal{D}) - Y(\mathcal{D} \setminus \{i\})$$

Under the assumption that $Y(S) = f(x_{\setminus \overline{S}})$, we add and subtract $f(x_{\setminus \mathcal{D}})$:

$$\text{TIE}(i) = f(x) - f(x_{\setminus \{i\}}) = f(x) - f(x_{\setminus \mathcal{D}}) - (f(x_{\setminus \{i\}}) - f(x_{\setminus \mathcal{D}}))$$

Now, we substitute $A_f(S) = f(x_{\setminus \overline{S}}) - f(x_{\setminus \mathcal{D}}) = \zeta(A_{\mu(f)})(S)$:

$$\text{TIE}(i) = A_f(\mathcal{D}) - A_f(\mathcal{D} \setminus \{i\}) = \zeta(A_{\mu(f)})(\mathcal{D}) - \zeta(A_{\mu(f)})(\mathcal{D} \setminus \{i\})$$

Expanding the definition of ζ and applying the inclusion-exclusion principle yields:

$$\text{TIE}(i) = \sum_{T \subseteq \mathcal{D}} A_{\mu(f)}(T) - \sum_{T \subseteq \mathcal{D}: i \notin T} A_{\mu(f)}(T) = \sum_{T \subseteq \mathcal{D}: i \in T} A_{\mu(f)}(T) = \sum_{T \subseteq \mathcal{D}: i \in T} \text{MI}(T)$$

□

ArchAttribute The ArchAttribute score for a set of features S can be represented as:

$$\phi(S) = \sum_{T \subseteq S} A_{\mu(f)}(T)$$

Proof. We start from the definition of ArchAttribute:

$$\phi(S) = f(x_{\setminus \overline{S}}) - f(x_{\setminus \mathcal{D}})$$

Substituting $A_f(S) = f(x_{\setminus \overline{S}}) - f(x_{\setminus \mathcal{D}}) = \zeta(A_{\mu(f)})(S)$ and expanding the definition of ζ yields:

$$\phi(S) = \zeta(A_{\mu(f)})(S) = \sum_{T \subseteq S} A_{\mu(f)}(T)$$

□

ArchDetect The ArchDetect score for a pair of features (i, j) can be represented as:

$$\overline{\omega}_{i,j} = \frac{1}{2h_i^2 h_j^2}\Big(\sum_{T \subseteq \mathcal{D}: i,j \in T} A_{\mu(f)}(T)\Big)^2 + (A_\mu(f)(\{i,j\}))^2$$

Proof. We start from the definition of ArchDetect:

$$\overline{\omega}_{i,j} = \frac{1}{2}\Big(\big(\frac{1}{h_i h_j}(f(x) - f(x_{\setminus \{i\}}) - f(x_{\setminus \{j\}}) + f(x_{\setminus \{i,j\}}))\big)^2 + \big(\frac{1}{h_i h_j}(f(x_{\setminus \overline{\{i,j\}}}) - f(x_{\setminus \overline{\{j\}}}) - f(x_{\setminus \overline{\{i\}}}) + f(x_{\setminus \mathcal{D}}))\big)^2\Big)$$

Substituting $A_f(S) = f(x_{\setminus \overline{S}}) - f(x_{\setminus \mathcal{D}})$, the first addend inside the outermost parentheses becomes:

$$\frac{1}{h_i^2 h_j^2}(A_f(\mathcal{D}) - A_f(\mathcal{D} \setminus \{i\}) - A_f(\mathcal{D} \setminus \{j\}) + A_f(\mathcal{D} \setminus \{i,j\}))^2$$

Similarly, the second addend becomes:

$$\frac{1}{h_i^2 h_j^2}(A_f(\{i,j\}) - A_f(\{i\}) - A_f(\{j\}))^2$$

Rewriting both addends in terms of $\zeta(A_\mu(f))$ and expanding the definition of ζ, we obtain:

$$\overline{\omega}_{i,j} = \frac{1}{2h_i^2 h_j^2}(\sum_{T \subseteq \mathcal{D}} A_{\mu(f)}(T) - \sum_{T \subseteq \mathcal{D}: i \notin T} A_{\mu(f)}(T) - \sum_{T \subseteq \mathcal{D}: j \notin T} A_{\mu(f)}(T) + \sum_{T \subseteq \mathcal{D}: i,j \notin T} A_{\mu(f)}(T))^2 + (A_\mu(f)(\{i,j\}))^2$$

Applying the inclusion-exclusion principle to the first addend yields:

$$\overline{\omega}_{i,j} = \frac{1}{2h_i^2 h_j^2}(\sum_{T \subseteq \mathcal{D}: i,j \in T} A_{\mu(f)}(T))^2 + (A_\mu(f)(\{i,j\}))^2$$

□